

Instruktion för instuderingsmaterial

Fakta

Kurs: Systemutvecklare C/C++ Extended 2024
Klass: SUVx24
Teknikområde: Versionshantering i Git

Learning Target

Versionshantering i Git

Hur fungerar versionshantering genom att kombinera systemen **Git** och **GitHub**?

Introduktion

När vi lärt oss hantera källkod lokalt med **Git** kan vi ta steget vidare och börja samarbeta kring gemensam källkod. Då använder många **GitHub** i kombination med **Git**, där **GitHub** är en molntjänst som låter dig spara och dela kod med andra. Då kan vi ha ett så kallade **remote repository** och ett **local repository**, som vi synkroniserar mellan.

Material

Här nedan hittar du ditt självstudiematerial. Du ska studera materialet i den ordning det listas för att du på bästa sätt ska tillgodogöra dig materialet.

Instruktion för instuderingsmaterial

Learning Target

Hur använder vi **Git** och **GitHub** tillsammans med **VS Code**? Hur uppstår *merge*-konflikter och hur löser vi dem?

Hur

Gör som förra instuderingen och fokusera på att läsa och titta på materialet nedan och vara påläst, men inte kunna allt utantill.

Teoretiskt

Använd länkarna nedan (både text och video) och svara på tillhörande frågor.

Praktiskt

Vi kommer på **Workshop 14** att

- Skapa ett individuellt **GitHub** konto där det gäller att välja en lämplig epostadress
- Använda **Git** och kopla ihop med **GitHub**.
- Integrera **Git** med **VS Code** med tillhörande plugin.

OBS! Du behöver inte göra något av ovanstående praktiska moment i förväg

Instruktion för instuderingsmaterial

Git merge conflict

Text

[Git merge conflict tutorial](#)

Läs artikeln och svara på:

- Hur uppstår *merge conflict* i Git?
- Vilka åtgärder kan du göra för att lösa konflikten?
- Finns det alternativa sätt att lösa konflikterna på?

Video

[Git Branching and Merging - Detailed Tutorial](#) (ca 55 min)

- Vad är motivationen till att använda en *branch*?
- Hur skapar vi en ny *branch* och börjar använda den?
 - Kan vi fortsätta arbeta på originalbranchen (master)
- Hur fungerar *merging*?
- Varför uppstår *merge conflict*?
- Vad innebär *feature workflow*?
 - Beskriv stegen i arbetsflödet.
- Studera den vanliga konfliktsituationen som uppstår i programmeringsprojekt med *feature workflow* och hur den löses.

Här är länken till det [Git and GitHub cheatsheet](#), som visas i videon.

Om du vill se de tidigare avsnitten till ovanstående video:

[Git and GitHub - 0 Experience to Professional in 1 Tutorial \(Part 1\)](#) (ca 37 min)

[Git and GitHub - 0 Experience to Professional in 1 Tutorial \(Part 2\)](#) (ca 56 min)